Working Paper No. 135

AD 658494

RELAXATION AND THE DUAL METHOD

IN MATHEMATICAL PROGRAMMING

by

ARTHUR GEOFFRION

March, 1968

OPERATIONS RESEARCH DIVISION
WESTERN MANAGEMENT SCIENCE INSTITUTE
University of California, Los Angeles

D D C
RECEIVED
MAY 7 1968
C

This document has been approved for public release and sale; its distribution is unlimited.

30

University of California

Los Angeles

Western Management Science Institute

Working Paper No. 135

"RELAXATION AND THE DUAL METHOD IN MATHEMATICAL PROGRAMMING"

by

Arthur Geoffrion

March, 1968

This is a revised and considerably expanded version of Working Paper No. 111, "Direct Reduction of Large Concave Programs," December, 1966. The support provided by the Ford Foundation under a Faculty Research Fellowship for the Academic year 1967-8 is gratefully acknowledged. Additional support was provided under a grant from the National Science Foundation and a contract with the Office of Naval Research.

ABSTRACT

The tactic of "relaxation" has often been used in one guise or another in order to cope with mathematical programs with a large number of constraints, some or all of which may be only implicitly available. By "relaxation" we mean the solution of a given problem via a sequence of smaller problems that are relaxed in that some of the inequality constraints are temporarily ignored. Relaxation has been used primarily in the context of linear programming, but in this paper we examine a version that is valid for a general class of concave programs. Constraints are *dropped* as well as added from relaxed problem to relaxed problem. A specialization to the completely linear case is shown to be equivalent to Lemke's Dual Method. This result permits some pertinent inferences to be drawn from the extensive computational experience available for the (primal) Simplex Method. Other matters pertaining to computational efficacy are discussed. An interpretation of relaxation in terms of the dual in the nonlinear case is also established. The optimal multipliers generated by successive relaxed problems turn out to comprise a sequence of improving feasible solutions to the minimax dual. When interpreted in this way, it becomes apparent that relaxation corresponds to just the opposite tactic--which we call "restriction"--applied to the dual problem. Restriction is an equally interesting and useful tactic in its own right, and its main features are outlined.

I. INTRODUCTION

Quite often an optimization problem with some inequality constraints possesses one or more of the following properties:

(1) prior knowledge is available concerning which of the constraints might be active at an optimum solution;

(2) there are so many constraints that the dimension limits of coded algorithms for available computers are exceeded;

(3) some of the constraints are available only implicitly, and can be generated in explicit form only at substantial expense.

Property (1) may hold when a variant of the problem has been solved before, or when the problem is amenable to physical or mathematical insight. Property (2) is the seemingly ubiquitous bane of practical applications. And property (3), usually in conjunction with property (2), is a frequent consequence of mathematical manipulationsof a more natural problem formulation.

For problems such as these a rather obvious "relaxation" tactic comes to mind for use in conjunction with any algorithm that would be applicable were it not for properties (2) or (3): solve a relaxed version of the given problem that ignores some of the inequality constraints; if the resulting solution satisfies all of the ignored constraints then it must be optimal in the original problem, but otherwise generate and include one or more violated constraints in the relaxed problem and reoptimize it; continue to generate and add violated constraints in this fashion until the original problem has been solved[1].

[1] G.B. Dantzig [6] is largely responsible for popularizing this tactic in the context of linear programming. He called it "the method of additional restraints" for handling "secondary constraints". See also Thompson, Tonge and Zionts [25].

This tactic seems quite promising if (as is usually the case) only a fairly small proportion of all inequality constraints is actually binding at an optimal solution of the original problem, provided that reasonably efficient mechanisms are available for identifying violated constraints and reoptimizing the relaxed problems. A useful improvement involves dropping amply satisfied constraints of the relaxed problem from time to time, but this must be done so as not to destroy the inherent finiteness of the procedure.

It is interesting to observe that Lemke's Dual Method [17] can be interpreted as a procedure for implementing the improved tactic within the specialized context of linear programming. Curiously this conspicuous interpretation seems never to have been explicitly stated and proved in the subsequent literature, although it is certainly part of the "folklore" of linear programming and has been used in one form or another by several authors[2]. As a result of this gap in the literature, it would seem that the pedagogy and even development of mathematical programming has suffered unnecessarily. Primal-motivated methods using such tactics[3] and dual methods are rarely exhibited in their proper relation to one another, and it has seldom been recognized that computational experience with variants of truly primal methods tells us something about the behavior of methods using corresponding variants of relaxation. The purpose of this note is to help smoothe over this hiatus in the literature.

[2] See Balinski [1], Charnes, Cooper and Miller [4], Gomory [12], and Gomory and Hu [13].

[3] See, e.g., Benders [3], Cheney and Goldstein [5], Dantzig, Fulkerson and Johnson [7], Gomory [11], Kelley [16], Ritter [19], Stone [23], and Van Slyke and Wets [26].

In sec. II we formally state a version of relaxation that permits constraints to be dropped from, as well as added to, the relaxed problems. Termination in a finite number of iterations is easily shown for a general class of concave programs. In sec. III we establish, under a non-degeneracy assumption, that in the completely linear case a specialization of the relaxation tactic is equivalent to Lemke's Dual Method. Matters pertaining to computational efficacy are discussed in the following section. A number of inferences are drawn, with the help of the result of sec. III, from available computational experience with variants of the Simplex Method. In the fifth and final section we establish an enlightening interpretation of relaxation in terms of the dual problem in the nonlinear case. It turns out that the optimal multipliers generated by successive relaxed problems comprise a sequence of improving feasible solutions to the minimax dual. When interpreted in this way, it becomes apparent that relaxation applied to the (original) primal problem corresponds to just the opposite tactic--which we call "restriction"--applied to the dual. Restriction is an equally interesting and useful tactic in its own right, and we conclude the paper with an outline of its main features.

II. STATEMENT AND PROOF OF THE RELAXATION TACTIC

Let f, $g_1,\ldots,g_m$ be concave functions on a non-empty convex set $X \subseteq R^n$, and define $M = \{1,2,\ldots,m\}$. The problem

$$\text{(P)} \qquad \text{Maximize}_{x \in X} \ f(x) \ \text{ subject to } \ g_i(x) \geq 0, \ i \in M$$

will be converted to a finite sequence of smaller problems of the form

$$(P_S) \qquad \text{Maximize}_{x \in X} \ f(x) \ \text{ subject to } \ g_i(x) \geq 0, \ i \in S \subseteq M.$$

Assume that a subset S^o is known such that (P_{S^o}) admits an optimal solution x^{S^o} (with $f(x^{S^o}) < \infty$), and assume further that (P_S) admits an optimal solution whenever it admits a feasible solution and its maximand is bounded above on the feasible region. For these assumptions to hold it is of course sufficient, but not necessary, that X be compact and all functions continuous (one may enforce boundedness, if necessary, by using a "regularization" artifice). Under these assumptions, we shall show that the following tactic is well-defined and terminates in a finite number of steps.

<u>Relaxation</u>

<u>Step 0</u>: Put $\overline{f} = \infty$ and $S = S^o$, where S^o is any subset of M such (P_{S^o}) admits a finite optimal solution.

<u>Step 1</u>: Solve (P_S) for an optimal solution x^S if one exists; if none exists (i.e., (P_S) is infeasible), then terminate with the message "(P) infeasible". If $g_i(x^S) \geq 0$ for all $i \in M-S$, terminate with the message "x^S is an optimal solution of (P)"; otherwise, go to Step 2.

Step 2: Put v equal to any subset of M that includes at least one constraint violated[4] by x^S. If $f(x^S) < \bar{f}$, replace S by $E \cup v$, where $E = \{i \in S: g_i(x^S) = 0\}$, and $\bar{f}$ by $f(x^S)$; otherwise (i.e., if $f(x^S) = \bar{f}$), replace S by $S \cup v$. Return to Step 1.

This tactic simply goes from one relaxed problem to the next by adding at least one constraint that is violated at an optimal solution of the current relaxed problem, while deleting the amply satisfied constraints so long as the value of the objective function is decreasing. Eventually a relaxed problem is encountered that is either infeasible, in which case (P) obviously must be infeasible, or has an optimal solution that is also feasible in (P), in which case that solution obviously must solve (P).

To show that the relaxed problems which arise are either infeasible or admit an optimal solution, in view of our assumptions it is enough to show inductively that the sequence $<f^S>$ is non-increasing, where f^S is the supremum of the maximand of (P_S) (let $f^S = -\infty$ if (P_S) is infeasible). Certainly $f^{S \cup v} \leq f^S$, and $f^{E \cup v} \leq f^E$. We assert that $f^E = f^S$, which yields the desired monotonicity of $<f^S>$. This assertion is an easy consequence of

Lemma 1.1: Let x^S be optimal for (P_S). If $g_j(x^S) > 0$, where $j \in S$, then x^S is also optimal for (P_{S-j}).

Proof: Certainly $f^{S-j} \geq f(x^S)$. Suppose that $f^{S-j} > f(x^S)$. Then

[4] Note the wide latitude in the choice of v. A common choice of v is to make it the index of the most violated constraint, but many other choice criteria are possible. See the discussion of sec. IV.

there exists a point x' feasible in (P_{S-j}) such that $f(x') > f(x^S)$. We may assume $g_j(x') < 0$, or else x' would contradict the optimality of x^S in (P_S). By the concavity of f and the g_i, $i \in S$, and the convexity of X, it follows that for λ positive but sufficiently small the point $\lambda x' + (1-\lambda)x^S$ is feasible in (P_S). But then $f(\lambda x'+(1-\lambda)x^S) \geq \lambda f(x') + (1-\lambda)\ f(x^S) > f(x^S)$, which contradicts the definition of x^S. Hence $f^{S-j} = f(x^S)$, and x^S must be optimal for (P_{S-j}).

Thus far we have shown that the tactic is well-defined and that the sequence $<f(x^S)>$ is non-increasing. Since Step 2 only deletes amply satisfied constraints from S (before adding v) when $f(x^S)$ has just decreased, it follows from the finiteness of the number of possible subsets of M that $<f(x^S)>$ can remain constant for only a finite number of consecutive iterations. Again appealing to the finiteness of the number of possible trial sets, we see that finite termination is established.

<u>Theorem 1.1</u>: Relaxation terminates in a finite number of steps with either (a) an optimal solution of (P), or (b) the identification of a subset of the constraints of (P) that are collectively infeasible over X. Moreover, in case (a) a non-increasing sequence $<f(x^S)>$ of upper bounds on the optimal value of (P) is obtained.

It is worth emphasizing again at this point that relaxation is a tactic and no more. It is not a computational procedure for solving (P) until it is applied in conjunction with an algorithm for solving the relaxed problems (P_S). However, let not its utter simplicity in the mathematical sense belie its usefulness.

Other variants of this tactic for convex programming have been given by Geoffrion [8 and 9], Oettli [18], Sethi [21], and Takeuti [24]. See also Cheney and Goldstein [5] for an application and proof of similar tactics to problems with an infinite number of constraints (cf. footnote 9).

III. RELATION TO THE DUAL METHOD

The fact that a feasible solution of (P) is not obtained until the final step, and that $\langle f(x^S)\rangle$ is monotone decreasing to the optimal value of (P), suggests the adjective "dual" in describing relaxation. In this section we shall show that relaxation can be specialized in a natural way so as to be equivalent to Lemke's Dual Method [17] when (P) is a linear program[5]. A more general dual interpretation is suggested in the final section.

Let $f(x) = cx$, $g_i(x) = x_i$, $M = \{1,\dots,n\}$, and $X = \{x: Ax = b\}$ hold for (P), where A is $m_1 \times n$. The Dual Method is initiated with some set B^o of variables designated as "basic" which yields, from the "reduced costs" of an associated tabular representation of (P) (see below), a feasible solution of the dual to (P). Assuming that the successive feasible solutions to the dual are non-degenerate, we shall prove

<u>Theorem 2.1</u>: If S^o is taken as $M-B^o$ and v always as the most violated constraint, then the set of non-basic variables at the ν^{th} iteration of the Dual Method coincides with E at the ν^{th} iteration of relaxation, and the ν^{th} basic solution coincides with the ν^{th} x^S.

It is necessary to give a brief rendering of the Dual Method in order to establish the notation used in the proof. More complete details may be found, for example, in [15] or [17].

Problem (P) can be restated as one of maximizing z subject to $x \geq 0$ and the following equality constraints stated as a tableau $(m_1+1$ by $n+2)$ of detached coefficients:

[5] See also Beale's "method of leading variables" [2], developed independently but nearly equivalent to Lemke's Dual Method."

z	x	= 1
1	-c	0
C	A	b

At any given iteration there is specified a collection B of m_1 basic variables such that A_B^{-1} exists, where A_B is formed by extracting columns from A according to B, and such that $\bar{c} \equiv (c_B A_B^{-1})A$ .. $- c \geq 0$, where c_B is similarly formed by extraction according to B. Moreover, the equality constraints are re-expressed as:

z	x	= 1
1	$(c_B A_B^{-1})A - c$	$c_B A_B^{-1} b$
0	$A_B^{-1} A$	$A_B^{-1} b$

If $\bar{b} \equiv A_B^{-1} b \geq 0$, then it is easily shown that an optimal solution of (P) is at hand: put $x_j = 0$ for j non-basic and the basic variable x_{B_i} (corresponding to the i^{th} row) equal to $\bar{b}_i$. If $\bar{b} \not\geq 0$, then let $\bar{b}_r$ be the most negative component (actually, <u>any</u> negative component will dc) and test make sure that at least one component $\bar{a}_{rj}$ of the matrix $A_B^{-1} A$ is negative for some non-basic j (if none is negative, it can be shown that (P) is infeasible). Let k be defined so that

$$\frac{\bar{c}_k}{\bar{a}_{rk}} = \text{Maximum } \left\{ \frac{\bar{c}_j}{\bar{a}_{rj}} : j \text{ non-basic and } \bar{a}_{rj} < 0 \right\},$$

and pivot on the element $\bar{a}_{rk}$ to obtain the detached coefficient array corresponding to the new set of basic variables $(B - B_r + k)$ (x_k is called the "entering," and x_{B_r} the "exiting" basic variable). If $\bar{c}_k > 0$ then $c_B A_B^{-1} b$ strictly decreases, and in any event the new $\bar{c}$ is also non-negative. The assumption of dual non-degeneracy means that $\bar{c}_j > 0$ for all non-basic j at each iteration, and can always be enforced by arbitrarily small perturbations of the problem data.

We are now in a position to make three key observations about the Dual Method. The first can be found essentially in Charnes, Cooper and Miller [4, p. 7ff].

<u>Lemma 2.1</u>: At any iteration of the Dual Method, the current basic solution is the unique optimal solution of (P_S) with S equal to the current set of non-basic variables $(S = M\text{-}B)$.

<u>Proof:</u> The current basic solution is certainly feasible in (P_{M-B}). To show that it is optimal, by the Dual Theorem of linear programming it suffices to display a feasible solution to the dual of (P_{M-B}) with the same value of the objective function. One has only to verify, using $\bar{c} \geq 0$, that $(c_B A_B^{-1})$ is such a dual solution. Uniqueness of the optimal solution of (P_{M-B}) follows from the assumed non-degeneracy of the dual.

<u>Lemma 2.2</u>: If the Dual Method terminates because (P) is infeasible (i.e., if $\bar{b}_r < 0$ and $\bar{a}_{rj} \geq 0$ for all non-basic j at some tableau), then (P_S), with S equal to the current set of non-basic variables plus B_r, is infeasible.

<u>Proof</u>: By the Dual Theorem of linear programming, it is enough to show that the dual of (P_S) is feasible and has an unbounded optimum. It

may be verified that $(c_B A_B^{-1}) + \theta(A_B^{-1})_{r.}$, where $(A_B^{-1})_{r.}$ is the r^{th} row of A_B^{-1} $(\bar{b}_r < 0)$, is feasible in the dual for all $\theta \geq 0$ and achieves an arbitrarily small value of the dual objective as $\theta \to \infty$.

<u>Lemma 2.3</u>: At any non-terminal iteration of the Dual Method, if x_k is the entering basic variable then $x_k > 0$ in the next basic solution.

<u>Proof</u>: The definition of the pivot operation implies that $x_k = (\bar{b}_r/\bar{a}_{rk})$ in the next basic solution. By selection, $\bar{b}_r < 0$ and $\bar{a}_{rk} < 0$.

<u>Proof of Th. 2.1</u>: The proof proceeds by induction on ν. At $\nu = 1$, S^o has been taken as $M\text{-}B^o$, the initial set of non-basic variables. Lemma 2.1 asserts that (P_{S^o}) has a unique solution x^{S^o}. Hence x^{S^o} must be the initial basic solution. Since $x_j^{S^o} = 0$ by definition for all non-basic j, $E = S^o$. Thus the assertion is true for $\nu = 1$.

Assume that the assertion is true for the $\nu_o{}^{th}$ iteration of the Dual Method. Either the $\nu_o{}^{th}$ iteration is terminal because (P) has been solved, or is terminal because (P) has been found to be infeasible, or is not terminal. In the first case, relaxation also terminates with an optimal solution of (P). In the second case, by Lemma 2.2 the next relaxed problem encountered is infeasible and therefore terminal. Consider now the third case. We shall show that the assertion of the theorem holds at the next iteration by detailing the operation of relaxation starting at Step 2 of the current iteration.

Dual non-degeneracy implies that $f(x^S)$ decreases strictly at each iteration. Hence the trial set to be used at the $(\nu_o+1)^{st}$

iteration of relaxation is $E \cup B_r$, where x_{B_r} is the most negative component of the current x^S. It follows from Lemmas 2.1 and 2.3 that $(P_{E \cup B_r})$ has a unique solution, and that all components indexed by $E \cup B_r$ vanish in this solution except for x_k, which is strictly positive. The assertion of the theorem now follows immediately.

IV. COMPUTATIONAL EFFICACY

Let us turn now to questions concerning computational efficacy. We have already mentioned in sec. I three properties which strongly encourage - if not demand - the use of tactics such as the present one. But this is not to say that computational success will necessarily be achieved if these properties hold. Computational success probably depends more on the following three conditions:

(a) only a fairly small proportion of all inequality constraints should actually be binding at an optimal solution of (P);

(b) a reasonably efficient mechanism must be available for identifying suitable violated constraints given a trial solution of (P);

(c) a reasonably efficient mechanism must be available for reoptimizing the reduced problem at each iteration.

That condition (a) often holds has frequently been mentioned (and exploited via similar tactics) for various special problem classes; for example, by Dantzig, Fulkerson and Johnson [7] for a linear programming equivalent of the traveling salesman problem, by Dantzig [6] for oil refinery problems, by Charnes, Cooper and Miller [4] for bounded variables and warehouse-type problems, and by Van Slyke and Wets [26] for optimal control and stochastic programming problems.

In fact it is easy to see that condition (a) will always hold for linear programs with a large number of inequality constraints relative to the number of structural variables, since an optimal solution occurs at an extreme point of the feasible region[6]. The same line of reasoning does not apply to nonlinear problems, but from curvature considerations and common experience it appears that condition (a) holds even more strongly than in the linear case.

Condition (b) is least troublesome if the inequality constraints are reasonable in number and explicitly available, for then there is no difficulty in implementing any reasonable criterion for the choice of v. Most commonly v is taken to be the most violated constraint, but many other criteria are possible. There is little theoretical or empirical evidence to distinguish these criteria from one another in terms of relative effectiveness. In view of the result of sec. III, however, we can perhaps draw some tentative inferences based on experience with the purely linear case. Extensive experiments have been carried out (e.g., [27]) comparing alternative rules for selecting pivotal columns in the usual Simplex Method for linear programming. This is actually the same as comparing analagous rules for choosing a singleton v with relaxation applied to the dual problem. Results indicate that while the "most violated constraint" rule may not be best in terms of minimizing the number of required iterations, other plausible rules can be expected to be consistently better by no more

[6] It should be noted that one could argue (cf. Smith and Orchard-Hays [22] and Stone [23]) for the usefulness in linear programming of tactics of the present sort even in the absence of condition (a).

than a factor of two or so. An example of a somewhat better rule is the so-called "greatest-change" rule, which for the present tactic amounts to choosing i in M-S to maximize the decrease in the optimal value of the next relaxed problem. Unfortunately such a rule is likely to be expensive to implement for a nonlinear problem. Choosing v to be the most violated constraint typically leads, in the linear case, to a number of iterations equal to about twice the number of variables. Results pertinent to the choice of v when more than one constraint index is allowed are available from experiments with the "suboptimization" tactic [27, p. 190]. It was observed[7], for example, that taking v to consist of the five most violated constraints reduced the number of iterations by a factor of two as compared with the single most violated constraint rule. Of course this increases the amount of computation required to solve each relaxed problem, but with the product form of the Simplex Method there is a significant net benefit in terms of total computing time. It is not known to what extent experience such as this for the linear case is a useful guide for the choice of v in the nonlinear case.

Condition (b) is more troublesome when the constraints are vast in number or only implicitly available. In this case concern over the best criterion for the choice of v is often all but irrelevant, since none but the simplest criteria can be implemented at reasonable computational cost. Sometimes only a few violated constraints are inexpensively available each time the relaxed problem is solved, and it

[7] Again we invoke a dual interpretation of the primal algorithm.

is indicated that they be used whether or not they satisfy any global criterion. This is the case, for example, with Dantzig, Fulkerson and Johnson's problem [7],with Gomory's integer programming algorithms [12, p. 133], and with Kelley's cutting-plane method [16][8]. On other occasions one can implement the "most violated constraint" criterion by solving a subsidiary optimization problem (or several smaller subsidiary problems, should special structure cause the constraints to partition naturally into several groups). This was the prescription of Cheney and Goldstein [5] in most of their algorithms[9]. For Benders [3] the subsidiary problem took the form of a linear program, and for Gomory and Hu [13] it took the form of network flow problems. For other special structures the subsidiary problem of finding the most violated constraint might assume the form of a dynamic program, or an integer program, etc.;

[8] To formally view Kelley's method as an instance of relaxation, one may represent the relevant portion of the set $\{x:g_i(x) \geq 0,\ i \in M\}$ in (P) by the intersection of an infinite number of containing half-spaces. (P) can thus be written as the problem of maximizing $f(x)$ subject to x in X and $G(x') + \gamma_{x'}^{t}(x-x') \geq 0$ for all x' in X such that $G(x') \leq 0$, where $G(x) = \text{Min}\,\{g_1(x),\ldots, g_m(x)\}$ and $\gamma_{x'}$ is a subgradient of G at x' (if $g_{i_o}(x') = G(x')$ and g_{i_o} is differentiable, then one can take $\gamma_{x'}$ as the gradient of g_{i_o} at x'). Kelley's choice of v corresponds to the constraint $G(x^S) + \gamma_{x}^{t}S(x-x^S) \geq 0$.

[9] See especially algorithms I, II, and IV. The minimand of each problem is the supremum of a collection of linear functions. To view this work in the present context, each minimand should be expressed as a collection of constraints using an additional variable and the least upper bound definition of a supremum. An interesting historical sidelight mentioned by Cheney and Goldstein is that the roots of their algorithms, and hence of relaxation, date back to E. Remez's work on polynomial approximation published in 1934.

examples of such algorithms are readily available (see Balinski [1] and Gomory [12])if we interpret primal algorithms with column-generating techniques as dual algorithms for the dual problem with row-generating techniques. Gilmore and Gomory [10, p. 877] have reasoned along these lines to establish a connection between their computational experience with the cutting-stock problem and previous experience with Gomory's integer programming algorithms. They conclude that in large linear programming problems computation times are likely to be long or erratic when v is a singleton chosen more or less blindly from the violated constraints, as opposed to the choice of v as the most violated constraint.

Condition (c) is met by the usual post-optimality techniques for adding additional constraints if (P_S) is a linear program. These techniques typically involve an iteration or two of the Dual Method, although they can be viewed in purely primal terms (consider the additional constraints as functions to be maximized until their values reach 0)[10]. The latter view is an appropriate one to use in the general nonlinear case, since it leads to a fairly easy modification of most primal nonlinear programming algorithms applicable to (P_S) and takes advantage of the availability of a feasible and optimal solution to the previous relaxed problem.

[10] Alternatively, one can parametrically deform (in any of several ways) each relaxed problem into the next one.

V. DUAL INTERPRETATION: RESTRICTION

The utter simplicity of relaxation certainly makes its interpretation in terms of (P) completely transparent. In the purely linear case there is no difficulty interpreting relaxation in terms of the dual of (P) as well, since the Dual Method amounts to the ordinary Simplex Method applied to the dual problem. In this section we establish an interpretation of relaxation in terms of the dual of (P) for the nonlinear case. It is of considerable interest that relaxation applied to (P) corresponds to a "restriction" tactic applied to the dual. Restriction is a useful tactic in its own right, with a rationale and justification paralleling that of relaxation in many respects.

The natural dual problem[11] associated with (P) is

$$(D) \qquad \underset{\lambda_i \geq 0,\ i\epsilon M}{\text{Minimize}} \left[\sup_{x\epsilon X} f(x) + \sum_{i\epsilon M} \lambda_i g_i(x) \right].$$

We assert that the sequence of optimal multipliers associated with the g_i constraints of the successive relaxed problems -- which can be guaranteed to exist under various mild qualifications -- constitutes a sequence of improving feasible solutions to (D). Let us denote the optimal multipliers for (P_S) by λ_i^S, $i\epsilon S$. Since λ_i^S is necessarily non-negative, the feasibility of these multipliers in (D) is immediate (take $\lambda_i^S = 0$ for $i\epsilon M-S$). By the saddlepoint condition characterizing (x^S, λ^S), moreover, we have

$$f(x^S) = \text{Max}_{x\epsilon X} \ f(x) + \sum_{i\epsilon M} \lambda_i^S \ g_i(x).$$

[11] A thorough discussion of modern nonlinear duality theory is given by Rockafellar [20]. We assume here at least a passing familiarity with the main concepts and results.

That $< \lambda^S >$ is an improving sequence offeasible solutions of (D) is thus confirmed by the result of sec. II that $< f(x^S) >$ is a non-increasing sequence.

This result leads one to ask whether there is a natural rationale for relaxation when viewed as a method for solving the dual problem. The answer is affirmative: relaxation applied to (P) amounts to a "restriction" tactic applied to (D). To explain this assertion, it will be more enlightening to explain "restriction" as applied to (P) rather than to (D). The reader can then understand our assertion in light of the fact that $g_i(x^S)$ plays the role of the dual variable associated with λ_i in (D); it can be shown that $(g_1(x^S),\ldots,g_m(x^S))$ is a subgradient of the minimand of (D) evaluated at λ^S (defined as above for $S \subseteq M$).

Let us now briefly consider restriction, the opposite of relaxation, in the context of (P). Again (P) is converted into a sequence of simpler problems, but now the simpler problems are restricted instead of relaxed. Each is of the form

$$(Q_S) \quad \underset{x \in X}{\text{Maximize}} \quad f(x) \quad \text{subject to} \quad \begin{aligned} &g_i(x) = 0, \ i \in S \subseteq M \\ &g_i(x) \geq 0, \ i \in M\text{-}S, \end{aligned}$$

where S is a subset of the constraint indices. In order that (Q_S) should be a concave program we require g_i to be linear for $i \in S$; so we may as well assume that g_i is linear for all $i \in M$ by incorporating any nonlinear constraints into X. Usually the constraints $g_i(x) \geq 0$, $i \in M$, will include the customary non-negativity constraints on the variables $(g_i(x) = x_i)$. In this case the variables indexed by

S vanish in (Q_S); thus if S is populous relative to M, (Q_S) is much more tractible than (P).

Assume that (Q_S) admits an optimal solution whenever it is feasible and its maximand is bounded above on the feasible region, and that in this event optimal multipliers associated with the g_i constraints are available. Then the following tactic is well-defined and terminates in a finite number of steps.

Restriction

Step 0 Put $\bar{f} = -\infty$ and $S = S^o$, where S^o is any subset of M such that (Q_{S^o}) is feasible (such a subset fails to exist if and only if (P) is infeasible).

Step 1 Solve (Q_S) for an optimal solution x^S (if the maximand of (Q_S) is unbounded above, then the same is obviously true of (P)). If the optimal multipliers μ_i^S associated with constraints $g_i(x) = 0$ $(i \in S)$ are all non-negative, then terminate with the message "x^S is an optimal solution of (P)"; otherwise, go to Step 2.

Step 2 Put v equal to any subset of S that includes at least one constraint in S for which $\mu_i^S < 0$. If $f(x^S) > \bar{f}$, replace S by $S \cup E-v$ where $E = \{i \in M-S: g_i(x^S) = 0\}$; otherwise (i.e., if $f(x^S) = \bar{f}$), replace S by $S-v$. Return to Step 1.

Note that constraints can enter the restricted set S, as well as leave it, so long as $f(x^S)$ is increasing. Clearly each trial solution x^S is feasible in (P), and $< f(x^S) >$ is a non-decreasing sequence.

It is easy to see that an appropriate specialization of this tactic to the linear case is equivalent to the ordinary Simplex Method. The set S then corresponds at each iteration to the current non-basic variables.

The circumstances in which restriction is an appealing tactic are precisely those mentioned in sec. I, if we read "variables" for "constraints" in the three properties of (P) mentioned there. For example, restriction is appealing when the number of variables is very great, or when the problem data corresponding to many of the variables are available only implicitly unless substantial expense is incurred. The circumstances in which restriction is likely to be computationally effective are analagous to those discussed in the previous section for relaxation. The analogue of condition (b), for instance, is that there must be a reasonably efficient mechanism at Step 2 for identifying variables in S whose corresponding multipliers are negative. Indeed, this is exactly what "column-generation" schemes for large-scale linear programming are all about. See the surveys by Balinski [1] and Gomory [12] for lucid discussions of such schemes.

That restriction is most highly developed in the context of linear programming should not obscure its applicability in the non-linear case. An outstanding example of the use of restriction for large structured nonlinear programs is Rosen's convex partition programming algorithm (in [14]), where it is used subsequent to a "partitioning" of the variables.

Restriction and relaxation, although opposites of one another, are by no means incompatible. The author has indicated elsewhere ([8] and [9]) how the two tactics can be employed simultaneously. The reduced problems then become simpler still than (P_S) or (Q_S), but assurance of finite termination requires somewhat more intricate control. The computational advantages of such a combined approach can be dramatic[12].

[12] A.M. Geoffrion, "Constrained Maximum Liklihood Estimation of Several Stochastically Ordered Distributions," The RAND Corporation, forthcoming.

REFERENCES

1. Balinski, M.L., "On Some Decomposition Approaches in Linear Programming," Notes for Recent Mathematical Advances in Operations Research, Engineering Summer Conferences, University of Michigan, 1966.

2. Beale, E.M.L., "An Alternative Method for Linear Programming," Proc. Camb. Phil. Soc., 50 (1954), 513-523.

3. Benders, J.F., "Partitioning Procedures for Solving Mixed-Variables Programming Problems," Numerische Mathematik, 4 (1962), 238-252.

4. Charnes, A., W.W. Cooper and M. Miller, "Dyadic Programs and Subdual Methods," Naval Research Logistics Quarterly, 8 (1961), 1-23.

5. Cheney, E.W. and A.A. Goldstein, "Newton's Method for Convex Programming and Tchebycheff Approximations," Numerische Mathematik, 1 (1959), 253-268.

6. Dantzig, G.B., "Upper Bounds, Secondary Constraints, and Block Triangularity in Linear Programming," Econometrica, 23 (1955), 174-183.

7. Dantzig, G., R. Fulkerson and S. Johnson, "Solution of a Large-Scale Traveling Salesman Problem," Operations Research, 2 (1954), 393-410.

8. Geoffrion, A.M., "Reducing Concave Programs with Some Linear Constraints," SIAM Journal on Applied Mathematics, 15 (1967), 653-664.

9. Geoffrion, A.M., "A Markovian Procedure for Concave Programming with Some Linear Constraints," forthcoming in the Proceedings of the Fourth International Conference on Operational Research, Cambridge, Mass., August 29-September 2, 1966.

10. Gilmore, P.C. and R.E. Gomory, "A Linear Programming Approach to the Cutting Stock Problem - Part II," Operations Research, 11 (1963), 863-888.

11. Gomory, R.E., "An Algorithm for Integer Solutions to Linear Programs," in [14].

12. Gomory, R.E., "Large and Nonconvex Problems in Linear Programming," in Experimental Arithmetic, High Speed Computing and Mathematics, Proceedings of Symposia in Applied Mathematics, Vol. XV, Americal Mathematical Society, 1963, 125-139.

13. Gomory, R.E. and T.C. Hu, "An Application of Generalized Linear Programming to Network Flows," SIAM Journal on Appl. Math., 10 (1962), 260-283.

14. Graves, R.L. and P. Wolfe, eds., Recent Advances in Mathematical Programming, McGraw-Hill, New York, 1963.

15. Hadley, G., Linear Programming, Addison-Wesley, Reading Mass., 1962.

16. Kelley, J.E., Jr., "The Cutting-Plane Method for Solving Convex Programs," SIAM Journal on Applied Mathematics, 8 (1960), 703-712.

17. Lemke, C.E., "The Dual Method of Solving the Linear Programming Problem," Naval Research Logistics Quarterly, 1 (1954), 36-47.

18. Oettli, K.W., "Die Rolle der wesentlichen Restriktionen bei einem konvexen Programm," Juris-Verlag, Zurich, 1964.

19. Ritter, K., "A Decomposition Method for Linear Programming with Coupling Constraints and Variables," MRC Report No. 739, Mathematics Research Center United States Army, University of Wisconsin, April 1967.

20. Rockafellar, R.T., Convex Analysis, Princeton University Press, Princeton, New Jersey, to appear in 1968.

21. Sethi, V.K., "Solution of a Class of Programming Problems," Sankhya, 25 (1963), Series B, 341-344.

22. Smith, D.M. and W. Orchard-Hays, "Computational Efficiency in Product Form LP Codes," in [14].

23. Stone, J.J., "The Cross-Section Method: An Algorithm for Linear Programming," P-1460, The RAND Corporation, Santa Monica, California, September 16, 1958.

24. Takeuti, K., "Note on Convex Programming" (Japanese), Keiei-Kagaku, 8 (1965), 155-162.

25. Thompson, G.L., F.M. Tonge and S. Zionts, "Techniques for Removing Nonbinding Constraints and Extraneous Variables from Linear Programming Problems," Management Science, 12 (1966), 588-608.

26. Van Slyke, R.M. and R.J.B. Wets, "L - Shaped Linear Programs with Applications to Optimal Control and Stochastic Programming," ORC 66-17, Operations Research Center, University of California at Berkeley, June 1967.

27. Wolfe, P. and L. Cutler, "Experiments in Linear Programming," in [14].

DOCUMENT CONTROL DATA - R&D

(Security classification of title body of abstract and indexing annotation must be entered when the overall report is classified)

1 ORIGINATING ACTIVITY *(Corporate author)* Western Management Science Institute	2a REPORT SECURITY CLASSIFICATION Unclassified 2b GROUP

3 REPORT TITLE

Relaxation and the Dual Method in Mathematical Programming

4 DESCRIPTIVE NOTES *(Type of report and inclusive dates)*

5 AUTHOR(S) *(Last name, first name, initial)*

Geoffrion, Arthur

6 REPORT DATE March, 1968	7a TOTAL NO OF PAGES 25	7b NO OF REFS

8a CONTRACT OR GRANT NO Nonr 233(75) b PROJECT NO c d	9a ORIGINATOR'S REPORT NUMBER(S) Working Paper No. 135 9b OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*
10 AVAILABILITY/LIMITATION NOTICES Distribution of this Document is unlimited	Western Management Science Institute University of California Los Angeles, California 90024
11 SUPPLEMENTARY NOTES	12 SPONSORING MILITARY ACTIVITY

13 ABSTRACT

The tactic of "relaxation" has often been used in one guise or another in order to cope with mathematical programs with a large number of constraints, some or all of which may be only implicitly available. By "relaxation" we mean the solution of a given problem via a sequence of smaller problems that are relaxed in that some of the inequality constraints are temporarily ignored. Relaxation has been used primarily in the context of linear programming, but in this paper we examine a version that is valid for a general class of concave programs. Constraints are dropped as well as added from relaxed problem to relaxed problem. A specialization to the completely linear case is shown to be equivalent to Lemke's Dual Method. This result permits some pertinent inferences to be drawn from the extensive computational experience available for the (primal) Simplex Method. Other matters pertaining to computational efficacy are discussed. An interpretation of relaxation in terms of the dual in the nonlinear case is also established. The optimal multipliers generated by successive relaxed problems turn out to comprise a sequence of improving feasible solutions to the minimax dual. When interpreted in this way, it becomes apparent that relaxation corresponds to just the opposite tactic -- which we call "restriction" -- applied to the dual problem. Restriction is an equally interesting and useful tactic in its own right, and its main features are outlined.